

Accounting for Classical Hardware in the Control of Quantum Devices

I.N. Hincks,^{1,2} C.E. Granade,^{3,2} T.W. Borneman,^{3,2} and D.G. Cory^{4,2,5,6}

¹*Department of Applied Math, University of Waterloo, Waterloo, ON, Canada*

²*Institute for Quantum Computing, Waterloo, ON, Canada*

³*Department of Physics, University of Waterloo, Waterloo, ON, Canada*

⁴*Department of Chemistry, University of Waterloo, Waterloo, ON, Canada*

⁵*Perimeter Institute for Theoretical Physics, Waterloo, ON, Canada*

⁶*Quantum Information Science Program, Canadian Institute for Advanced Research, Toronto, ON, Canada*

(Dated: September 30, 2014)

High fidelity coherent control of quantum systems is critical to building quantum devices and quantum computers. We provide a general optimal control framework for designing control sequences that account for hardware control distortions while maintaining robustness to environmental noise. We demonstrate the utility of our algorithm by presenting examples of robust quantum gates optimized in the presence of nonlinear distortions. We show that nonlinear classical controllers do not necessarily incur additional computational cost to pulse optimization, enabling more powerful quantum devices.

PACS numbers:

The ability to coherently control the dynamics of quantum systems with high fidelity is a critical component of the development of modern quantum devices, including quantum computers [1], actuators [2, 3], and sensors [4–6] that push beyond the capabilities of classical computation and metrology. In recent years, quantum computation has presented a compelling application for quantum control, as high-fidelity control is essential to implement quantum information processors that achieve fault-tolerance [7–9].

The performance of numerically optimized quantum gates in laboratory applications strongly depends on the accuracy of the system model used to approximate the response of the experimental system to the applied control sequence. Here we develop a general framework whereby classical control hardware components are modelled explicitly, such that their effect on a quantum system can be computed and compensated for using numerical optimal control theory (OCT) [10] algorithms to optimize control sequences. Control sequences designed using OCT algorithms, such as the GRAdient Ascent Pulse Engineering (GRAPE) [11] algorithm, can be made robust to a wide variety of inhomogeneities, pulse errors and noise processes [12–14]. These methods are also easily extended [15–18] to other applications and may be integrated into other protocols [19]. Recently, it was demonstrated how a model of linear distortions of the control sequence, such as those arising from finite bandwidth of the classical control hardware, may also be integrated into OCT algorithms [20–22].

Here, we improve and generalize those results beyond linear kernels to any operation that smoothly maps a list of control steps to a classical field seen by the quantum system. Importantly, our framework naturally allows for robustness against uncertainties and errors due to classical control hardware. We begin developing our method generally, without making assumptions about the device of interest, so that our results may be broadly applicable

to a wide range of quantum devices. We briefly discuss how our theory is easily applied to any linear distortion, and then in more detail, demonstrate with numerics how nonlinearities in control hardware, such as those found in strongly-driven superconducting resonators used for pulsed electron spin resonance (PESR) [23–25], may be included in OCT algorithms.

With this goal in mind, we briefly review the problem of controlling a quantum system [26]. Given a system Hamiltonian

$$H(t) = H_0 + \sum_{l=1}^L q_l(t) H_l \quad (1)$$

where H_0 is the internal Hamiltonian and $\{H_l\}_{l=1}^L$ are the control Hamiltonians, how do we choose the envelopes $\{q_l(t)\}_{l=1}^L$ such that at time T we effect the total unitary U_{target} ? It will be clear that the framework we construct will be compatible with not only this specific unitary control problem, but all similar problems such as state to state transfers, expectation values over static distributions, open system maps, etc.

The functions $\{q_l(t)\}_{l=1}^L$ seen by the quantum system represent a distorted version of what was input to the classical hardware. Since we are ultimately interested in doing numerics, we begin by discretizing the time domain and therefore model all relevant hardware by what we will call a *discretized distortion operator*. This is a function $g : \mathbb{R}^N \otimes \mathbb{R}^K \rightarrow \mathbb{R}^M \otimes \mathbb{R}^L$ which takes an input pulse sequence, \vec{p} , with some associated time step dt , and outputs a distorted version of the pulse, $\vec{q} = g(\vec{p})$, with an associated time step δt . \vec{p} is the pulse as generated by the experimenter's computer, and \vec{q} is the pulse generating the Hamiltonian seen by the quantum system, as illustrated in Figure 1.

The integers N and M are the number of input and output time steps respectively, and K and L are the number of input and output control fields respectively. In the

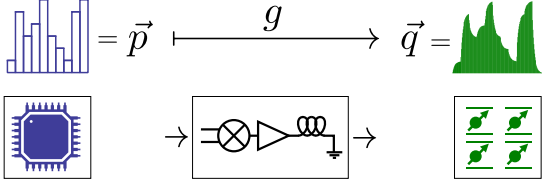


FIG. 1: A cartoon depicting the action of the distortion operator g on the input pulse \vec{p} .

case of quadrature control of a qubit, $K = L = 2$. We omit subscripts on the time steps dt and δt for notational simplicity; uniform time discretization is not required. Typically, we will have $\delta t < dt$ to allow for an accurate simulation of the quantum system. The condition $M \cdot \delta t = N \cdot dt$ need not hold, for example, $M \cdot \delta t > N \cdot dt$ will be useful when the distortion has a finite ringdown time.

The discretized distortion operator g will often derive from a *continuous distortion operator* $f : \mathbf{L}_1(\mathbb{R}, \mathbb{R}^K) \rightarrow \mathbf{L}_1(\mathbb{R}, \mathbb{R}^L)$ which takes a continuous input pulse $\alpha(t)$ and outputs a distorted pulse $\beta(t) = f[\alpha](t)$. The discretized version is obtained by composing f on either side by a discretization and dediscretization operator, $g = f_1 \circ f \circ f_2$.

We can incorporate the distortion operator g into standard techniques from optimal control theory. In particular, consider the unitary objective function,

$$\Phi[\vec{q}] = \left| \text{Tr} \left(U_{\text{target}}^\dagger \prod_{m=1}^M e^{-i\delta t(H_0 + \sum_{l=1}^L q_{m,l} H_l)} \right) \right|^2 / d^2, \quad (2)$$

used in the GRAPE algorithm [11]. Penalties can be added to this basic objective function in order to demand that the solution admit certain properties. For instance, penalty functions have been used to ensure robustness to control noise and limited pulse fluence [13, 27–29] or to ensure that undesired subspaces are avoided [30, 31].

Now we include the effect of our hardware by modifying the objective function to compose with the distortion operator,

$$\Phi_g[\vec{p}] = \Phi \circ g(\vec{p}). \quad (3)$$

Using the multivariable chain rule, we compute the gradient of Φ_g to be

$$\nabla_{\vec{p}}(\Phi_g) = \nabla_{g(\vec{p})}(\Phi) \cdot J_{\vec{p}}(g) \quad (4)$$

$$[J_{\vec{p}}(g)]_{m,l,n,k} = \frac{\partial g_{m,l}}{\partial p_{n,k}} \quad (5)$$

where the dot represents a contraction over the indices m and l , and where $J_{\vec{p}}(g)$ is the Jacobian of g at \vec{p} . Though evaluating $\nabla_{g(\vec{p})}(\Phi)$ naively would require simulating the action of $M \times L$ pulses, the GRAPE algorithm

[11] provides an expression for this gradient in terms of the timestep unitaries that are already computed,

$$\frac{\partial \Phi}{\partial q_{m,l}} = -2 \text{Re} [\langle P_m | i\delta t H_l X_m \rangle \langle X_m | P_m \rangle], \quad (6)$$

where $P_m := \left(\prod_{i=m+1}^M U_i^\dagger \right) U_{\text{target}}$, $X_m := \prod_{i=m}^1 U_i$ and where $U_i(\vec{q}) = \exp(-i\delta t[H_0 + \sum_{l=1}^L q_{m,l} H_l])$. Therefore if we can compute the Jacobian $J_{\vec{p}}(g)$, we can then compute the total gradient of Φ . The rest of the algorithm follows as described in the original GRAPE [11].

Since the cost of evaluating g will typically not grow more than polynomially with the number of qubits, the computational cost of the optimization effectively remains unchanged from standard GRAPE, as it is still dominated by the cost of computing the M matrix exponentials.

Our first example is the continuous distortion operator given by the convolution with an $L \times K$ kernel $\phi(t)$,

$$\beta(t) = f(\alpha)(t) = (\phi \star \alpha)(t) = \int_{-\infty}^{\infty} \phi(t - \tau) \cdot \alpha(\tau) d\tau. \quad (7)$$

The convolution kernel ϕ models any distortion that can be described by a linear differential equation, such as a simple exponential rise time, control line crosstalk, or the transfer function of the control hardware [20, 21, 32, 33]. We compute the discretized distortion operator to be

$$q_{m,l} = \sum_{n=1, k=1}^{N,K} \left(\int_{(n-1)dt}^{ndt} \phi_{l,k}((m-1/2)\delta t - \tau) d\tau \right) p_{n,k}. \quad (8)$$

where we see that it acts as a linear map,

$$\vec{q} = g(\vec{p}) = \tilde{\phi} \cdot \vec{p}, \quad (9)$$

where we are contracting over the n and k indices with the components of the tensor $\tilde{\phi}$ given by the integrals

$$[\tilde{\phi}]_{m,l,n,k} = \int_{(n-1)dt}^{ndt} \phi_{l,k}((m-1/2)\delta t - \tau) d\tau. \quad (10)$$

The Jacobian matrix is simply given by $J_{\vec{p}}(g) = \tilde{\phi}$ which is independent of the pulse \vec{p} .

As a more involved example we consider a quantum system being controlled by a tuned and matched resonator circuit [34] with nonlinear circuit elements (Figure 2). Nonlinear resonators are used in a variety of applications, including superconducting qubits for quantum information processing [35], microwave kinetic inductance detectors for astronomy [36], and increasing inductive detection sensitivity in magnetic resonance [37]. Often, however, electronics controlling quantum systems are operated in their linear regime to avoid complications resulting from nonlinearity [25]. Avoiding nonlinearities requires reducing input power, leading to longer control sequences that reduce the number of quantum operations

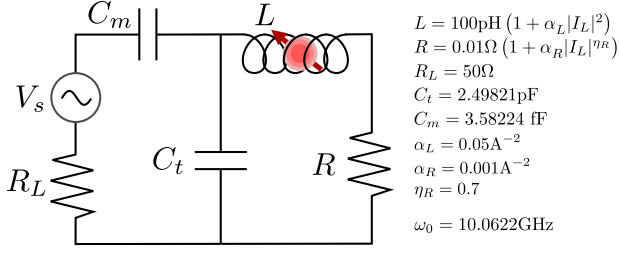


FIG. 2: A quantum system being controlled by the magnetic field produced by the inductor of a nonlinear resonator circuit. The ideal voltage source $V_s(t)$ is specified by the input undistorted pulse \tilde{p} , and the resulting current through the inductor, $I_L(t)$, is computed. The inductance and the resistance are both functions of the current passing through them. The form of the nonlinearity is chosen to be consistent with kinetic inductance.

that can be performed before the system decoheres. Additionally, limiting input power removes the natural robustness of high-power sequences to uncertainties in the environment achieved by strongly modulating the quantum system [38, 39].

If the circuit were linear, the distortion could be modelled as a convolution $\phi \star$ as discussed above. However, with nonlinear circuit elements present we must numerically solve the circuit's differential equation every time we wish to compute the distorted pulse [40].

For concreteness, but with no loss of generality, we work with an on-resonance qubit system whose Hamiltonian in the rotating frame, after invoking the rotating wave approximation, is

$$H = \frac{\delta\omega}{2}\sigma_z + (1 + \kappa) \left(\frac{\omega_x(t)}{2}\sigma_x + \frac{\omega_y(t)}{2}\sigma_y \right) \quad (11)$$

where $\delta\omega$ and κ represent off-resonance and control power errors, respectively.

The time evolution of the circuit shown in Figure 2 is governed by the third order differential equation

$$\frac{d}{dt} \begin{bmatrix} I_L \\ V_{C_m} \\ V_{C_t} \end{bmatrix} = \begin{bmatrix} -\frac{R}{L} & 0 & \frac{1}{L} \\ 0 & \frac{-1}{R_L C_m} & \frac{1}{R_L C_m} \\ \frac{-1}{C_t} & \frac{1}{R_L C_t} & \frac{1}{R_L C_t} \end{bmatrix} \begin{bmatrix} I_L \\ V_{C_m} \\ V_{C_t} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{V_s(t)}{R_L C_m} \\ \frac{V_s(t)}{R_L C_t} \end{bmatrix} \quad (12)$$

where the nonlinearities arise when the inductance, L , and resistance, R , are functions of the current passing through them [40, 41]. In the case of kinetic inductance, these nonlinearities take on the form

$$\begin{aligned} L &= L(I_L) = L_0(1 + \alpha_L |I_L|^2) \\ R &= R(I_R) = R_0(1 + \alpha_R |I_R|^\eta) \end{aligned} \quad (13)$$

where α_L , α_R and η are constants [42, 43]. Kinetic inductance leads to a reduction in the circuit resonance frequency, coupling, and quality factor with increasing power, as shown in Figure 3(a-b).

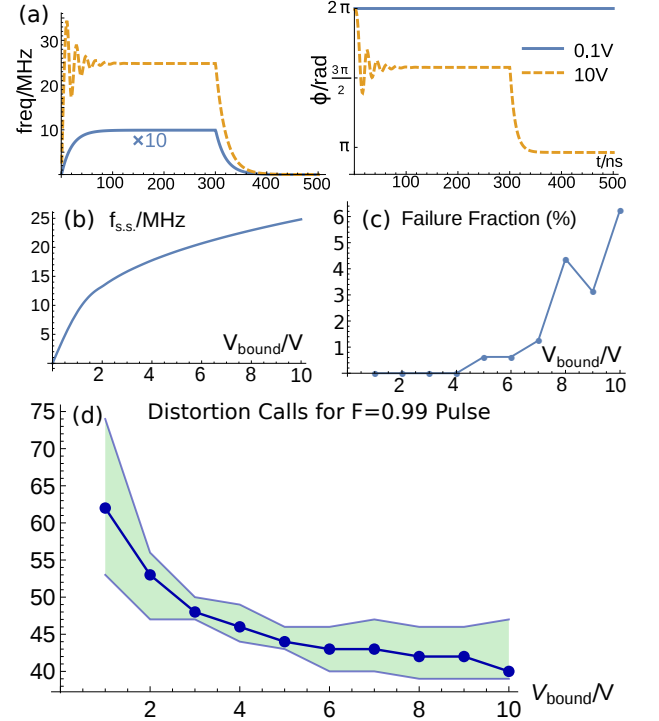


FIG. 3: (a) Response from the same resonator to a square forcing term with length 300ns in both a linear (0.1V) and nonlinear (10V) regime. The amplitude of the 0.1V pulse is multiplied by 10 to make it visible. (b) The steady state driving frequency as seen by the spins as a function of the voltage input to the resonator. (c) Out of 160 pulses searched for at each of 10 voltage bounds, the fraction that failed to reach $F = 0.99$ before the step size was effectively zero, and (d) the median number of calls made to the distortion function g along with the 16% and 84% quantiles during the gradient ascent for those pulses which did reach $F = 0.99$.

Since our Hamiltonian in Equation 11 is written in a frame rotating at the circuit resonance frequency in the linear-regime, it is convenient to write our differential equation in this frame. To this end, with the differential equation 12 shorthand as $\dot{\vec{y}}(t) = B(\vec{y}(t))\vec{y}(t) + V_s(t)\vec{b}$, introduce the complex change of variables $\vec{x}(t) = e^{-i\omega_0 t}\vec{y}(t)$. In this new frame, since $B(\vec{y}(t)) = B(\vec{x}(t))$, our dynamics become

$$\begin{aligned} \dot{\vec{x}}(t) &= (B(\vec{x}(t)) - i\omega_0 \mathbb{I})\vec{x}(t) + \tilde{V}_s(t)\vec{b} \\ &\equiv A(\vec{x}(t))\vec{x}(t) + \tilde{V}_s(t)\vec{b} \end{aligned} \quad (14)$$

where we have invoked the rotating wave approximation, and $\tilde{V}_s(t)$ is the rotating version of $V_s(t)$. Now the real and imaginary parts of the complex current in the rotating frame, $\tilde{I}_L(t) = e^{-i\omega_0 t} I_L(t)$, are proportional via a geometric factor to the control amplitudes appearing in the Hamiltonian,

$$\omega_x(t) \propto \text{Re}[\tilde{I}_L(t)] \quad \text{and} \quad \omega_y(t) \propto \text{Im}[\tilde{I}_L(t)]. \quad (15)$$

To compute the distortion $\vec{q} = g(\vec{p})$ caused by the resonator, we set the circuit's input voltage $\tilde{V}_s(t)$ to be the piecewise constant function with amplitudes coming from \vec{p} . To improve stiffness conditions, a small finite risetime may be added to the forcing term $\tilde{V}_s(t)$, which is equivalent to adding a low-pass filter to the ideal voltage source in the circuit. We can now solve the equations 14 for $\tilde{I}_L(t)$ using the NDSolve function in Mathematica 10, interpolate the results, and resample at a rate δt to determine the distorted pulse \vec{q} .

Since our distortion is nonlinear, the Jacobian of g will not be constant with respect to the input pulse \vec{p} . However, we may compromise the accuracy of the Jacobian in favour of taking a larger number of ascent steps that are still generally uphill by considering only the Jacobian at the zero pulse,

$$\left. \frac{\partial g_{m,l}}{\partial p_{n,k}} \right|_{\vec{p}} \approx [g(\epsilon \vec{e}_{n,k})/\epsilon]_{m,l}. \quad (16)$$

These quantities may be precomputed prior to gradient ascent and therefore only add a constant to the computation time. Exact partial derivatives may be computed for a cost that scales as $K \cdot N$ and whose implementation can be highly parallelized; see the Appendix for details.

In Figure 4, we show an example of a GRAPE-optimized pulse for $U = (\frac{\pi}{2})_x$, with the circuit of Figure 2 used as a distortion operator. There are 16 times steps of length 0.5ns shown as a solid red step function. The pulse has been made to be robust to static uncertainty in the Hamiltonian parameters $\delta\omega$ and γ and the nonlinearity parameter α_L . Since the circuit has a high quality factor, it would take many times the length of the pulse for the ringdown tail to decay to zero. We therefore utilize an active ringdown suppression scheme with three compensation steps of lengths 4ns, 2ns, and 1ns. This is a generalization of ringdown suppression in linear circuits [21, 44, 45] and is discussed in detail in the Appendix.

The inclusion of a distortion operator causes an alteration to the control landscape which might be expected to make finding optimal solutions more expensive. Therefore, a tradeoff between computational cost and gate time length might be anticipated. We perform a numerical

study to examine this relationship.

We bound the allowed input power to the resonator used by the GRAPE algorithm by 10 different voltages, 1 V to 10 V, where 1 V is on the edge of the linear regime, and 10 V is highly nonlinear. In analogy to the numerical control landscape experiments performed in Reference [46], for each of these bounds, we attempt to compute a fidelity $F = 0.99 (\frac{\pi}{2})_x$ pulse 160 times, with a different random initial guess each time. The total length of the pulse is set to $T_{\text{pulse}} = \frac{0.25}{f_{\text{s.s.}}}$ where $f_{\text{s.s.}}$ is the steady state driving frequency of the resonator at the corresponding voltage bound. The number of time steps is held constant at $N = 16$ for each trial. The gradient approximation from (C8) is used. On each trial, we count the number of times the distortion function g is called. The results are shown in Figure 3 where it is seen that the number of calls actually tends to decrease as the allowed nonlinearity is increased, indicating that the control landscape does not become more difficult to navigate.

In conclusion, we have presented an optimization framework that permits the design of robust quantum control sequences that account for general simulatable distortions by classical control hardware. We have demonstrated that even when distortions are nonlinear with respect to the input – using the particular example of a nonlinear resonator circuit – robust quantum control may still be achieved, and searching through the control landscape does not necessarily become more difficult. Thus, classical control devices may be operated in their high power regime to permit fast high fidelity quantum operations, increasing the number of gates that can be performed within the decoherence time of the quantum system.

Acknowledgments

Funding from Industry Canada, CERC, NSERC, Province of Ontario, CFI, and Mitacs is gratefully acknowledged. I.H. thanks Hamid Mohebbi, Daniel Puzoli and Osama Moussa for helpful discussions.

-
- [1] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien, *Nature* **464**, 45 (2010), ISSN 0028-0836, URL <http://www.nature.com/nature/journal/v464/n7285/full/nature08812.html>.
 - [2] J. S. Hodges, J. C. Yang, C. Ramanathan, and D. G. Cory, *Physical Review A* **78**, 010303 (2008), URL <http://link.aps.org/doi/10.1103/PhysRevA.78.010303>.
 - [3] T. W. Borneman, C. E. Granade, and D. G. Cory, *Physical Review Letters* **108**, 140502 (2012), URL <http://link.aps.org/doi/10.1103/PhysRevLett.108.140502>.
 - [4] P. Cappellaro, J. Emerson, N. Boulant, C. Ramanathan, S. Lloyd, and D. G. Cory, *Physical Review Letters* **94**, 020502 (2005), URL <http://link.aps.org/doi/10.1103/PhysRevLett.94.020502>.
 - [5] H. J. Mamin, M. Kim, M. H. Sherwood, C. T. Rettner, K. Ohno, D. D. Awschalom, and D. Rugar, *Science* **339**, 557 (2013), ISSN 0036-8075, 1095-9203, URL <http://www.sciencemag.org/content/339/6119/557>.
 - [6] J. M. Taylor, P. Cappellaro, L. Childress, L. Jiang, D. Budker, P. R. Hemmer, A. Yacoby, R. Walsworth, and M. D. Lukin, *Nature Physics* **4**, 810 (2008), ISSN 1745-2473, URL <http://www.nature.com/nphys/>

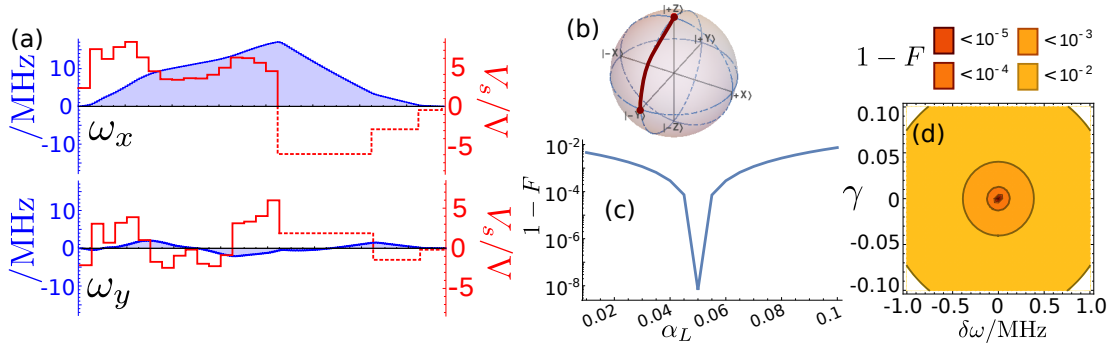


FIG. 4: (color online) (a) Example of a $\pi/2_x$ pulse generated for the matched nonlinear resonator circuit. The driving term (\vec{p}) is shown in red, while the distorted pulse (\vec{q}) is shown in blue. The dashed segments are the ringdown compensation steps. (b) The trajectory of the state $|0\rangle$ under this pulse is shown on the Bloch sphere, and (c-d) the average fidelity is plotted for different values of α_L , $\delta\omega$, and γ .

- journal/v4/n10/abs/nphys1075.html.
- [7] D. Gottesman, arXiv:0904.2557 (2009), URL <http://arxiv.org/abs/0904.2557>.
- [8] A. G. Fowler, A. M. Stephens, and P. Groszkowski, Physical Review A **80**, 052312 (2009), URL <http://link.aps.org/doi/10.1103/PhysRevA.80.052312>.
- [9] M. Gutierrez, L. Svec, A. Vargo, and K. R. Brown, Physical Review A **87**, 030302 (2013), URL <http://link.aps.org/doi/10.1103/PhysRevA.87.030302>.
- [10] L. S. Pontryagin, *Mathematical theory of optimal processes* (CRC Press, 1987), URL http://books.google.ca/books?hl=en&lr=&id=kwwzQF4cBVAC&oi=fnd&pg=PR11&dq=The+Mathematical+Theory+of+Optimal+Processes&ots=3nn1lwoc_9&sig=ndv_yiATkRVt8qJP-yNvOaf1Tb4.
- [11] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbruggen, and S. J. Glaser, Journal of Magnetic Resonance **172**, 296 (2005), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/B6WJX-4DXT7T6-6/2/e86f276f798a97858c5a9df62d5f50c0>.
- [12] K. Kobzar, S. Ehni, T. E. Skinner, S. J. Glaser, and B. Luy, Journal of Magnetic Resonance **225**, 142 (2012), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780712003126>.
- [13] V. D. M. Koroleva, S. Mandal, Y.-Q. Song, and M. D. Hrlimann, Journal of Magnetic Resonance **230**, 64 (2013), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780713000207>.
- [14] T. W. Borneman, M. D. Hrlimann, and D. G. Cory, Journal of Magnetic Resonance **In Press**, **Corrected Proof** (2010), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/B6WJX-511G1Y5-2/2/ddb8975ea0ab760e5bd8a26ce6ee2ab1>.
- [15] S. Mandal, V. D. M. Koroleva, T. W. Borneman, Y.-Q. Song, and M. D. Hrlimann, Journal of Magnetic Resonance **237**, 1 (2013), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780713002255>.
- [16] S. Mandal, T. W. Borneman, V. D. M. Koroleva, and M. D. Hrlimann, Journal of Magnetic Resonance **247**, 54 (2014), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780714002250>.
- [17] T. Schulte-Herbruggen, A. Sprl, N. Khaneja, and S. J. Glaser, Journal of Physics B: Atomic, Molecular and Optical Physics **44**, 154013 (2011), ISSN 0953-4075, URL <http://iopscience.iop.org/0953-4075/44/15/154013>.
- [18] M. H. Goerz, D. M. Reich, and C. P. Koch, New Journal of Physics **16**, 055012 (2014), ISSN 1367-2630, URL <http://iopscience.iop.org/1367-2630/16/5/055012>.
- [19] D. Egger and F. Wilhelm, Physical Review Letters **112**, 240503 (2014), URL <http://link.aps.org/doi/10.1103/PhysRevLett.112.240503>.
- [20] P. E. Spindler, Y. Zhang, B. Endeward, N. Gershernzon, T. E. Skinner, S. J. Glaser, and T. F. Prisner, Journal of Magnetic Resonance **218**, 49 (2012), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780712000705>.
- [21] T. W. Borneman and D. G. Cory, Journal of Magnetic Resonance **225**, 120 (2012), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780712003333>.
- [22] F. Motzoi, J. M. Gambetta, S. T. Merkel, and F. K. Wilhelm, Physical Review A **84**, 022307 (2011), URL <http://link.aps.org/doi/10.1103/PhysRevA.84.022307>.
- [23] O. Benningshof, H. Mohebbi, I. Taminiau, G. Miao, and D. Cory, Journal of Magnetic Resonance **230**, 84 (2013), ISSN 10907807, URL <http://europepmc.org/abstract/med/23454577>.
- [24] H. Malissa, D. I. Schuster, A. M. Tyryshkin, A. A. Houck, and S. A. Lyon, Review of Scientific Instruments **84**, 025116 (2013), ISSN 0034-6748, 1089-7623, URL <http://scitation.aip.org/content/aip/journal/rsi/84/2/10.1063/1.4792205>.
- [25] A. J. Sigillito, H. Malissa, A. M. Tyryshkin, H. Riemann, N. V. Abrosimov, P. Becker, H.-J. Pohl, M. L. W. Thewalt, K. M. Itoh, J. J. L. Morton, et al., Applied Physics Letters **104**, 222407 (2014), ISSN 0003-6951, 1077-3118, URL <http://scitation.aip.org/content/aip/journal/apl/104/22/10.1063/1.4881613>.
- [26] D. d'Alessandro, *Introduction to quantum control and dynamics* (CRC press, 2007), URL <http://books.google.ca/books?hl=en&lr=&id=e5M0id5enzQC&oi=fnd&pg=PP1&dq=quantum+control+textbook&ots=>

- Erc5LQa0Qf&sig=RfMjyw6-kNkBa_GY6ZfuSajPkbs.
- [27] D. Hocker, C. Brif, M. D. Grace, A. Donovan, T.-S. Ho, K. W. M. Tibbetts, R. Wu, and H. Rabitz, arXiv:1405.5950 [quant-ph] (2014), URL <http://arxiv.org/abs/1405.5950>.
 - [28] T. E. Skinner, T. O. Reiss, B. Luy, N. Khaneja, and S. J. Glaser, Journal of Magnetic Resonance **167**, 68 (2004), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780703004221>.
 - [29] K. Kobzar, T. E. Skinner, N. Khaneja, S. J. Glaser, and B. Luy, Journal of Magnetic Resonance **194**, 58 (2008), ISSN 1090-7807, URL <http://www.sciencedirect.com/science/article/pii/S1090780708001936>.
 - [30] J. P. Palao, R. Kosloff, and C. P. Koch, Physical Review A **77**, 063412 (2008), URL <http://link.aps.org/doi/10.1103/PhysRevA.77.063412>.
 - [31] M. M. Miller, D. M. Reich, M. Murphy, H. Yuan, J. Vala, K. B. Whaley, T. C. Calarco, and C. P. Koch, Physical Review A **84**, 042315 (2011), URL <http://link.aps.org/doi/10.1103/PhysRevA.84.042315>.
 - [32] D. D. Rife and J. Vanderkooy, Journal of the Audio Engineering Society **37**, 419 (1989), URL <http://www.aes.org/e-lib/browse.cfm?elib=6086>.
 - [33] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, et al., Nature **508**, 500 (2014), ISSN 0028-0836, URL <http://www.nature.com/nature/journal/v508/n7497/full/nature13171.html>.
 - [34] T. M. Barbara, J. F. Martin, and J. G. Wurl, Journal of Magnetic Resonance (1969) **93**, 497 (1991), ISSN 0022-2364, URL <http://www.sciencedirect.com/science/article/pii/0022236491900788>.
 - [35] R. J. Schoelkopf and S. M. Girvin, Nature **451**, 664 (2008), ISSN 0028-0836, URL <http://www.nature.com/nature/journal/v451/n7179/full/451664a.html>.
 - [36] P. K. Day, H. G. LeDuc, B. A. Mazin, A. Vayonakis, and J. Zmuidzinas, Nature **425**, 817 (2003), ISSN 0028-0836, URL <http://www.nature.com/nature/journal/v425/n6960/full/nature02037.html>.
 - [37] G. Bachar, O. Suchoi, O. Shtempluck, A. Blank, and E. Buks, Applied Physics Letters **101**, 022602 (2012), ISSN 0003-6951, 1077-3118, URL <http://scitation.aip.org/content/aip/journal/apl/101/2/10.1063/1.4734500>.
 - [38] E. M. Fortunato, M. A. Pravia, N. Boulant, G. Teklemariam, T. F. Havel, and D. G. Cory, The Journal of Chemical Physics **116**, 7599 (2002), ISSN 0021-9606, 1089-7690, URL <http://scitation.aip.org/content/aip/journal/jcp/116/17/10.1063/1.1465412>.
 - [39] M. A. Pravia, N. Boulant, J. Emerson, A. Farid, E. M. Fortunato, T. F. Havel, R. Martinez, and D. G. Cory, The Journal of Chemical Physics **119**, 9993 (2003), ISSN 0021-9606, 1089-7690, URL <http://scitation.aip.org/content/aip/journal/jcp/119/19/10.1063/1.1619132>.
 - [40] Maas, *Nonlinear Microwave and RF Circuits* (Artech House Publishers, Boston, MA, 2003), 2nd ed., ISBN 9781580534840.
 - [41] M. Tinkham, *Introduction to Superconductivity: Second Edition* (Dover Publications, Mineola, N.Y., 2004), second edition edition ed., ISBN 9780486435039.
 - [42] T. Dahm and D. J. Scalapino, Journal of Applied Physics **81**, 2002 (1996), ISSN 0021-8979, URL <http://scitation.aip.org/content/aip/journal/jap/81/4/10.1063/1.364056>.
 - [43] H. R. Mohebbi, O. W. B. Benningshof, I. a. J. Taminiau, G. X. Miao, and D. G. Cory, Journal of Applied Physics **115**, 094502 (2014), ISSN 0021-8979, 1089-7550, URL <http://scitation.aip.org/content/aip/journal/jap/115/9/10.1063/1.4866691>.
 - [44] M. Mehring, *High resolution NMR in solids* (Springer, 1976), URL <http://link.springer.com/content/pdf/10.1007/978-3-642-68756-3.pdf>.
 - [45] D. I. Hoult, Review of Scientific Instruments **50**, 193 (1979), ISSN 0034-6748, 1089-7623, URL <http://scitation.aip.org/content/aip/journal/rsi/50/2/10.1063/1.1135786>.
 - [46] K. Moore, M. Hsieh, and H. Rabitz, The Journal of Chemical Physics **128**, 154117 (2008), ISSN 0021-9606, 1089-7690, URL <http://scitation.aip.org/content/aip/journal/jcp/128/15/10.1063/1.2907740>.
 - [47] S. Gustavsson, O. Zwier, J. Bylander, F. Yan, F. Yoshihara, Y. Nakamura, T. P. Orlando, and W. D. Oliver, Physical Review Letters **110**, 040502 (2013), URL <http://link.aps.org/doi/10.1103/PhysRevLett.110.040502>.
 - [48] M. H. Levitt, *Spin Dynamics: Basics of Nuclear Magnetic Resonance* (Wiley, 2001), ISBN 9780471489221.

Appendix A: Examples of Common Distortions

The distortion formalism outlined in the main body need not be applied to only complicated systems; it is just as useful when applied to simple systems. At a high level, having a general framework allows for problems to be tackled systematically, allows for solutions found in one modality to be easily transferred to another, and reduces development overhead as the system evolves. In this section we outline some common, simple, but useful distortions, and show how they can be written down as discrete distortion operators.

1. Composition

First, it is worth noting the (perhaps obvious) fact that composing distortion operators is easily implemented in this framework. Suppose we have characterized the first half of our classical hardware with the discrete distortion operator $g_1 : \mathbb{R}^N \otimes \mathbb{R}^K \rightarrow \mathbb{R}^{N'} \otimes \mathbb{R}^{K'}$ and the second half with the operator $g_2 : \mathbb{R}^{N'} \otimes \mathbb{R}^{K'} \rightarrow \mathbb{R}^M \otimes \mathbb{R}^L$. We have been careful to make the domain of g_2 the same as the range of g_1 . Then the total distortion operator is given by the

composition

$$g = g_2 \circ g_1 : \mathbb{R}^N \otimes \mathbb{R}^K \rightarrow \mathbb{R}^M \otimes \mathbb{R}^L \quad (\text{A1})$$

$$\vec{p} \mapsto g_2(g_1(\vec{p})). \quad (\text{A2})$$

To find the Jacobian matrix of g at point \vec{p} we just need to use the multivariate chain rule,

$$J_{\vec{p}}(g) = J_{g_1(\vec{p})}(g_2) \cdot J_{\vec{p}}(g_1), \quad (\text{A3})$$

or in terms of indices,

$$[J_{\vec{p}}(g)]_{m,l,n,k} = \sum_{n'=1}^{N'} \sum_{k'=1}^{K'} [J_{g_1(\vec{p})}(g_2)]_{m,l,n',k'} [J_{\vec{p}}(g_1)]_{n',k',n,k}. \quad (\text{A4})$$

2. Transfer Functions and Convolutions

Linear electronic systems can be fully described by a *transfer function* $\Phi(\omega)$. This function gives a simple relationship between an input tone $X(\omega)$ at frequency ω and the resulting output tone $Y(\omega)$, namely $Y(\omega) = \Phi(\omega)X(\omega)$. The magnitude of $H(\omega)$ represents the gain or attenuation, and the argument represents the phase shift. Taking the inverse Fourier transform of this equation yields the convolution, $y(t) = (\phi \star x)(t)$, where there may be factors of 2π missing due to convention. The transfer function may be measured experimentally [20, 32, 47], or may be computed if a good model of the system is known. In the main body, the formula for a discrete convolution operator arising from a time domain transfer function ϕ is shown. Here, we derive it in slightly more detail.

To begin, the time domain version of the transfer function $\phi(t)$ results in the distortion operator f defined as

$$\beta(t) = f(\alpha)(t) = (\phi \star \alpha)(t) = \int_{-\infty}^{\infty} \phi(t - \tau) \cdot \alpha(\tau) d\tau. \quad (\text{A5})$$

Note that here $\phi(t)$ is a function whose values are $L \times K$ matrices. In the usual context where $K = L$ and the k' th output channel is mostly a distorted version of the k' th input channel, the the diagonals of $\phi(t)$ represent channel-wise distortions, and the off-diagonals represent cross contamination between channels.

We can explicitly write the discretization and dediscretization operators mentioned in the main text as

$$\begin{aligned} f_1 : \mathbf{L}_1(\mathbb{R}, \mathbb{R}^K) &\rightarrow \mathbb{R}^M \otimes \mathbb{R}^K \\ \beta &\mapsto (\beta(\delta t \cdot (1/2)), \dots, \beta(\delta t \cdot (M - 1/2))) \end{aligned} \quad (\text{A6})$$

$$\begin{aligned} f_2 : \mathbb{R}^N \otimes \mathbb{R}^L &\rightarrow \mathbf{L}_1(\mathbb{R}, \mathbb{R}^L) \\ (\vec{p}_1, \dots, \vec{p}_N) &\mapsto \sum_{n=1}^N \vec{p}_n \cdot \text{Top}(t - dt \cdot (n - 1/2)) \end{aligned} \quad (\text{A7})$$

where Top is the L -dimensional top hat function,

$$T(t) = \begin{cases} (1, 1, \dots, 1) & 0 \leq t < dt \\ (0, 0, \dots, 0) & \text{else,} \end{cases} \quad (\text{A8})$$

and the factors of $1/2$ appear so that we are sampling the midpoint of each step. We are also using the convention that an element of $\vec{p} \in \mathbb{R}^N \otimes \mathbb{R}^K$ is thought of as a vector $\vec{p} = (\vec{p}_1, \dots, \vec{p}_N)$ of vectors, where each $\vec{p}_k \in \mathbb{R}^K$. This means our discretized distortion operator will be $g = f_1 \circ f \circ f_2$. Discretizing the input we get

$$f(f_2(\vec{p}))(t) = \sum_{n=1}^N \int_{(n-1)dt}^{ndt} \phi(t - \tau) \cdot \vec{p}_n \, d\tau \quad (\text{A9})$$

$$\equiv \sum_{n=1}^N \phi_n(t) \cdot \vec{p}_n, \quad (\text{A10})$$

which we then discretize the output of, to get

$$[(f_1 \circ f \circ f_2)(\vec{p})]_{m,l} = \sum_{n=1}^N \sum_{k=1}^K [\phi_n((m-1/2)\delta t)]_{l,k} p_{n,k} \quad (\text{A11})$$

$$\equiv \sum_{n=1}^N \sum_{k=1}^K \phi_{m,l,n,k} p_{n,k}, \quad (\text{A12})$$

for all $1 \leq m \leq M$ and $1 \leq l \leq L$ where

$$\phi_{m,l,n,k} = [\phi_n((m-1/2)\delta t)]_{l,k} \quad (\text{A13})$$

$$= \int_{(n-1)\delta t}^{n\delta t} [\phi((m-1/2)\delta t - \tau)]_{l,k} d\tau \quad (\text{A14})$$

Letting $\tilde{\phi} \in \mathbb{R}^M \otimes \mathbb{R}^L \otimes \mathbb{R}^N \otimes \mathbb{R}^K$ be the tensor with entries $\phi_{m,l,n,k}$ gives

$$g(\vec{p}) = (f_1 \circ f \circ f_2)(\vec{p}) = \tilde{\phi} \cdot \vec{p} \quad (\text{A15})$$

as a compact representation of the discretized distortion operator, where the dot represents a contraction over the indices n and k .

The elements of the Jacobian matrix $J(g)$ are now easily computed as

$$[J(g)]_{m,l,n,k} = \frac{\partial(g(\vec{p}))_{m,l}}{\partial p_{n,k}} = \frac{\partial(\tilde{\phi} \cdot \vec{p})_{m,l}}{\partial p_{n,k}} = \frac{\partial \sum_{n'=1}^N \sum_{k'=1}^K \phi_{m,l,n',k'} p_{n',k'}}{\partial p_{n,k}} = \phi_{m,l,n,k} \quad (\text{A16})$$

so that

$$J(f_\phi) = \tilde{\phi}. \quad (\text{A17})$$

3. Finite Rise Times

A simple special case of the general convolution discussed in the previous subsection is a rise time acting independently on each control channel. This will cause the rising edge of a square input pulse to be smoothed over with an exponential of time constant τ , and the trailing edge to decay back to zero with an exponential of the same time constant. Such a transfer function arises, for example, from a simple RL circuit, where the time constant will be given by $\tau = L/R$.

Given a rise time τ_c^k acting independently on each of the $1 \leq x \leq K = L$ control channels gives the time-domain transfer function as

$$\phi_{l,k}(t) = \begin{cases} \frac{1}{\tau_c^k} e^{-t/\tau_c^k} & l = k \text{ and } t \geq 0 \\ 0 & \text{else} \end{cases} \quad (\text{A18})$$

which, when the integral from (A14) is performed and simplified (by Mathematica in this case), results in the discrete convolution

$$[\tilde{\phi}]_{m,l,n,k} = \begin{cases} \delta_{l,k} (e^{dt/\tau_c^k} - 1) e^{\frac{t_{n-1}-t'_m}{\tau_c^k}} & t_n < t'_m \\ \delta_{l,k} \left(1 - e^{\frac{t_{n-1}-t'_m}{\tau_c^k}} \right) & (t_n = t'_m) \vee ((t_n > t'_m) \wedge (n = 1 \vee t_{n-1} < t'_m)) \\ 0 & \text{else} \end{cases} \quad (\text{A19})$$

for each $1 \leq k \leq K = L$, where $t_n = n\delta t$ and $t'_m = (m-1/2)\delta t$. This is illustrated in Figure 5 with $K = L = 1$.

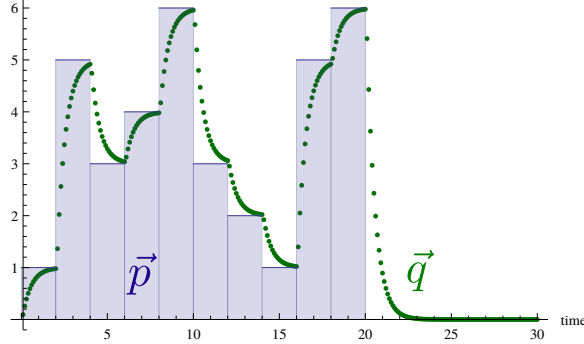


FIG. 5: An example of the application of a discrete convolution distortion g to an input pulse with $N = 10$ time steps. We have $dt = 2$, and the output space has 20 time steps per input time step, thus $\delta t = 0.1$.

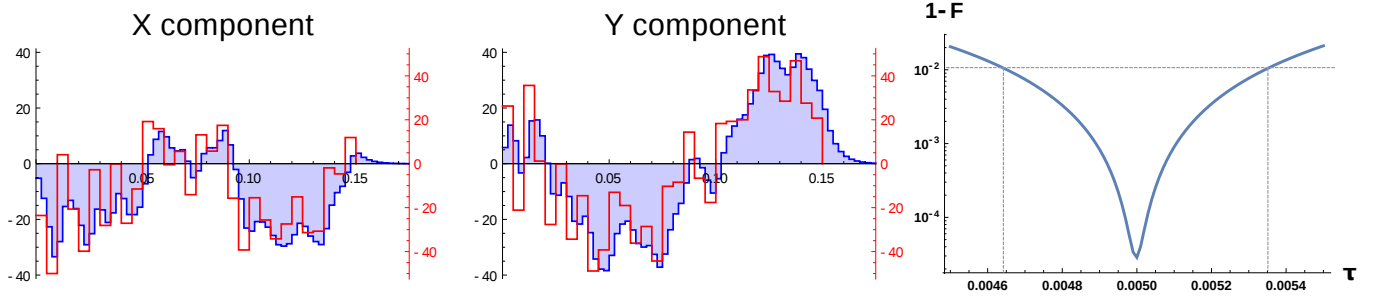


FIG. 6: The pulse envelope of a CNOT gate at an exponential rise time value of $\tau = 0.005$. The robustness curve, in terms of one minus average fidelity, is shown to the right as a function of τ .

a. Example of Robustness to Rise Time

We consider designing a CNOT gate for two qubits with an internal Hamiltonian

$$H = \frac{\omega_1}{2}\sigma_z^1 + \frac{\omega_2}{2}\sigma_z^2 + \frac{J}{4}(\sigma_x^1\sigma_x^2 + \sigma_y^1\sigma_y^2 + \sigma_z^1\sigma_z^2) \quad (\text{A20})$$

and control Hamiltonians

$$\{H_x = \sigma_x^1 + \sigma_x^2, \quad H_y = \sigma_y^1 + \sigma_y^2\} \quad (\text{A21})$$

where $\omega_1 = -2\pi \cdot 15$, $\omega_2 = +2\pi \cdot 15$, $J = 2\pi \cdot 50$, and the amplitudes of the control Hamiltonian are bounded by $2\pi \cdot 50$. This is the style of Hamiltonian found in liquid state NMR homonuclear samples [48]. We use $N = 30$ input time steps of length $dt = 0.005$, and $M = 2N + \lceil 10\tau/dt \rceil$ output time steps of length $dt/2$. Here, τ is the characteristic exponential rise time of both control channels, as defined in Section A 3.

To make the resulting pulse sequence robust against the value τ , we set the objective function of the optimization problem to be a convex combination of objective functions, each with a different value of τ , as explained in Section F. The results are shown in Figure 6, where it is seen that $F > 0.99$ is achieved in a region about $\pm 7\%$ around a nominal value of $\tau = 0.005$.

4. Crosstalk

Crosstalk is the phenomenon where a signal sent along one control channel is overheard by other control channels. As alluded to earlier, this can be fully accounted for (in the case of linear controllers) by the off-diagonal elements of the transfer function ϕ . This may be overkill as crosstalk can often be accurately modelled as one control line seeping

into each of the other control lines with attenuation factors that are constant in time. See Reference [33] for example, where crosstalk between five coupled superconducting qubits is observed.

To model this situation, we consider that our quantum system has I subsystems (or qubits) each with $L = K$ control channels. The Hamiltonian is given by

$$H = H_{\text{int}} + \sum_{i=1}^I \sum_{l=1}^L q_{i,l}(t) H_{i,l} \quad (\text{A22})$$

where H_{int} is the internal Hamiltonian, containing all coupling terms, and $q_{i,l}(t)$ and $H_{i,l}$ are the l^{th} control envelope and Hamiltonian of the i^{th} system. There are now a total of $I \cdot L$ controls, and so instead of indexing the control indices by single numbers, k and l , we index them by tuples, (i, k) and (i, l) . Since this distortion is independent of time, we set $\delta t = dt$ and $M = N$. Ideally, we would have $p_{n,(i,k)} = q_{n,(i,k)}$ representing the fact that the $(i, k)^{\text{th}}$ control signal is sent exactly to the $(i, k)^{\text{th}}$ Hamiltonian at each time step n . With crosstalk included, the $(i, k)^{\text{th}}$ Hamiltonian actually sees a linear combination of each of every control line,

$$q_{n,(i,l)} = \sum_{j=1}^I \sum_{k=1}^K \chi_{(i,l),(j,k)} p_{n,(j,k)}, \quad (\text{A23})$$

where $\chi_{(i,l),(j,k)}$ is the fraction of the $(j, k)^{\text{th}}$ control line seen on the $(i, l)^{\text{th}}$ control. More compactly,

$$\vec{q} = g(\vec{p}) = \chi \cdot \vec{p} \quad (\text{A24})$$

where the dot in this case represents contraction over the indices j and k .

The ideal χ tensor is $\chi_{(i,l),(j,k)} = \delta_{i,j} \delta_{l,k}$. As an example, if there are 4 qubits each with two controls, x and y, then in matrix format the ideal tensor reads

$$\chi_{\text{ideal}} = \begin{array}{c} \begin{array}{cc} & \begin{array}{cc} Q1 & Q2 & Q3 & Q4 \end{array} \\ \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} \end{array} \\ \begin{array}{cc} Q1 & \begin{array}{cc} x & y \end{array} \end{array} \left(\begin{array}{cc|cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array} \right) \end{array} \quad (\text{A25})$$

If we add a crosstalk term between adjacent qubits, where an x control only talks to adjacent x controls and similar for y controls, the tensor might look like

$$\chi_{\text{nearest neighbour}} = \begin{array}{c} \begin{array}{cc} & \begin{array}{cc} Q1 & Q2 & Q3 & Q4 \end{array} \\ \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} & \begin{array}{cc} x & y \end{array} \end{array} \\ \begin{array}{cc} Q1 & \begin{array}{cc} x & y \end{array} \end{array} \left(\begin{array}{cc|cc|cc|cc} 1 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0.3 & 0 & 0 & 0 & 0 \\ \hline -0.1 & 0 & 1 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0.15 & 0 & 1 & 0 & 0.4 & 0 & 0 \\ \hline 0 & 0 & -0.2 & 0 & 1 & 0 & 0.2 & 0 \\ 0 & 0 & 0 & -0.2 & 0 & 1 & 0 & 0.3 \\ \hline 0 & 0 & 0 & 0 & 0.23 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.7 & 0 & 1 \end{array} \right) \end{array} \quad (\text{A26})$$

As is clear from Equation A24, the Jacobian of this distortion operator is simply given by

$$J_{\vec{p}}(g) = \chi. \quad (\text{A27})$$

A pulse could be designed to be robust against errors in the crosstalk tensor by including a distribution over crosstalk tensors, or perhaps just a distribution over some of its values, dependently or independently, using the method described in Section F.

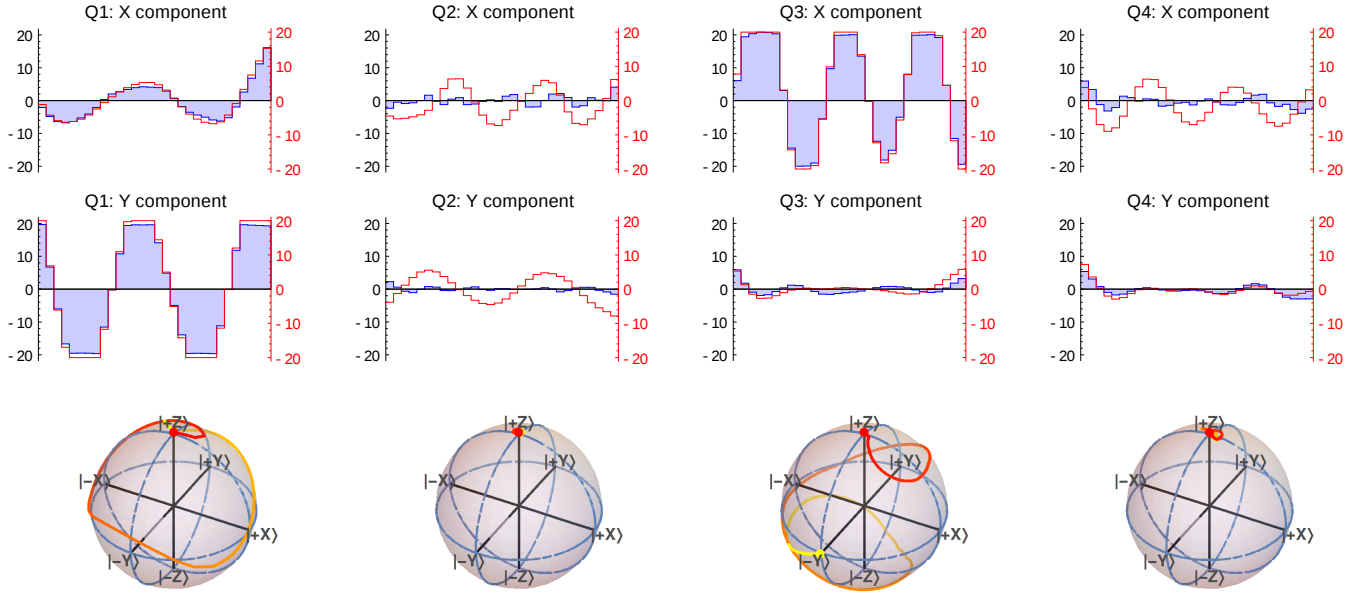


FIG. 7: (color online) The pulse envelopes and Bloch sphere trajectories of a $\pi/2)_x$ gate on the third qubit. The (unfilled) red curves represent the input pulse, and the (filled) blue curves represent the output pulse seen by the quantum system.

a. Crosstalk Example

We design a $\pi/2$ gate about x on the third of four qubits arranged in a line, with the other three qubits performing the identity operation. Each qubit has an x and y control, $\{H_{i,x} = \sigma_x^i, H_{i,y} = \sigma_y^i\}$, and the internal Hamiltonian is given by

$$H = \sum_{|i-j|=1} \omega_{ij} \sigma_z^i \sigma_z^j \quad (\text{A28})$$

where $\omega_{ij} = 2\pi \cdot 20\text{MHz}$ and the input control amplitudes are limited to $2\pi \cdot 40\text{MHz}$. We use a crosstalk tensor

$$\chi = \begin{matrix} & \begin{matrix} Q1 & Q2 & Q3 & Q4 \end{matrix} \\ \begin{matrix} Q1 \\ Q2 \\ Q3 \\ Q4 \end{matrix} & \begin{matrix} x & y & x & y & x & y & x & y \\ \begin{pmatrix} x & y \\ 1 & 0 & 0.3 & 0.001 & 0.05 & 0 & 0.001 & 0 \\ 0 & 1 & 0 & 0.1 & 0 & 0.01 & 0 & 0.001 \end{pmatrix} \end{matrix} \end{matrix} \quad (\text{A29})$$

Using this crosstalk distortion tensor, a pulse with average fidelity $F = 0.9999$ was found and is shown in [Figure 7](#).

Appendix B: The Rotating Frame of the Circuit

A spin in a large static magnetic field $\gamma B_0 = \omega_0$ with a transverse time dependent field $\gamma B_1(t) = 2\Omega(t)$ will evolve under the Hamiltonian

$$H = \frac{\omega_0}{2} \sigma_z + 2 \frac{\Omega(t)}{2} \sigma_x. \quad (\text{B1})$$

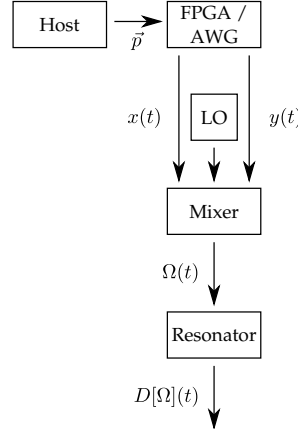


FIG. 8: Configuration of the microwave mixing components in relation to pulse distortion operators.

Since ω_0 is taken to be the dominant term, in analogy to a wide range of experimental settings, it is helpful to enter the rotating frame generated by $H_{\text{rot}} := \omega_r \sigma_z / 2$, where $\omega_r := [\omega_0 + \delta\omega]$. In doing so, we will suppose that

$$\begin{aligned}\Omega(t) &= \omega_1(t) \cos(\omega_r t + \phi(t)) \\ &= \omega_1(t) \cdot \frac{e^{it(\omega_r t + \phi(t))} + e^{-it(\omega_r t + \phi(t))}}{2},\end{aligned}$$

representing that Ω is produced by mixing a modulating signal with an oscillator at $\omega_0 + \delta\omega$ (see Figure 8). We will later relate this model to the in-phase and quadrature control fields.

In the frame of H_{rot} , the effective Hamiltonian H_{eff} is given by

$$\begin{aligned}H_{\text{eff}}(t) &= e^{+iH_{\text{rot}}t} H(t) e^{-iH_{\text{rot}}t} - H_{\text{rot}} \\ &= \Omega(t) e^{+iH_{\text{rot}}t} \sigma_x e^{-iH_{\text{rot}}t} - \delta\omega \sigma_z \\ &= \Omega(t) [\cos(\omega_r t) \sigma_x - \sin(\omega_r t) \sigma_y] - \delta\omega \sigma_z.\end{aligned}$$

Discarding the terms which oscillate at $2\omega_r$ (that is, the rotating wave approximation), we can rewrite this in terms of ω_1 and ϕ instead,

$$H_{\text{eff}}(t) = \frac{\omega_1(t)}{2} [\cos(\phi(t)) \sigma_x + \sin(\phi(t)) \sigma_y] - \delta\omega \sigma_z.$$

Since Ω is often produced by mixing, as noted above, we can also represent the rotating frame control using two real control fields $\omega_x(t)$ and $\omega_y(t)$,

$$\omega_x(t) = \text{Re}[\omega_1(t) e^{i\phi(t)}] \quad \omega_y(t) = \text{Im}[\omega_1(t) e^{i\phi(t)}]. \quad (\text{B2})$$

Using these control fields, we match the definition of H_{eff} in the main body, given as Equation 11.

This rotating frame can be analogously extended into the circuit dynamics as well. The nutation frequency of the spins is proportional to the magnetic field generated by the inductor via the gyromagnetic ratio, which is in turn proportional to the current passing through the inductor, thus

$$\Omega(t) = \kappa I_L(t), \quad (\text{B3})$$

where the exact value of κ depends on the relevant geometry. This along with Equation B2 leads us to the relationships

$$\begin{aligned}\omega_x(t) &= \text{Re}[\omega_1(t) e^{i\phi(t)}] = \text{Re}[\Omega(t) e^{-i\omega_r t}] = \kappa \text{Re}[I_L e^{-i\omega_r t}] = \kappa \text{Re}[\tilde{I}_L] \\ \omega_y(t) &= \text{Im}[\omega_1(t) e^{i\phi(t)}] = \text{Im}[\Omega(t) e^{-i\omega_r t}] = \kappa \text{Im}[I_L e^{-i\omega_r t}] = \kappa \text{Im}[\tilde{I}_L],\end{aligned} \quad (\text{B4})$$

where $\tilde{I}_L(t) = e^{-i\omega_1 t} I_L(t)$ and we have ignored pieces rotating at $2\omega_r$ in the calculation. We recover the expression in the main body, such that by solving the differential equation $\dot{\vec{x}}(t) = A(\vec{x})\vec{x} + \tilde{V}_s(t)\vec{b}$ for a complex driving function $\tilde{V}_s(t)$, we can find the rotating-frame unitary action $U(t) = \text{T exp}(-i \int_0^t H_{\text{eff}}(t))$ of that pulse.

Appendix C: The Discretized Distortion Operator Due to a Resonator Circuit

1. Definition of the Distortion Operator

We define the distortion operator $g : \mathbb{R}^N \otimes \mathbb{R}^2 \rightarrow \mathbb{R}^M \otimes \mathbb{R}^2$ corresponding to the non-linear resonator circuit shown in the main body. Note that the input to g will have units of volts, and the output of g will have units of Hz. We use a uniform input discretization time dt and a uniform output discretization time δt . Given an input pulse $\vec{p} \in \mathbb{R}^N \otimes \mathbb{R}^2$, we define the complex vector $\tilde{p} \in \mathbb{C}^N$ by $\tilde{p}_n = p_{n,1} + ip_{n,2}$. Setting $n(t) = \lceil \frac{t}{dt} \rceil$, we define

$$\alpha(t) = \tilde{p}_{n(t)} \quad (C1)$$

in the case where we add no rise time to the forcing term, or

$$\alpha(t) = \tilde{p}_{n(t)-1} + (\tilde{p}_{n(t)} - \tilde{p}_{n(t)-1})(1 - e^{-\frac{t-n(t)dt}{\tau_r}}). \quad (C2)$$

in the case where we include a finite rise time on the forcing with timescale τ_r . Note that we must have $\tau_r \ll dt$ for the function α to be (approximately) continuous. This limitation could easily be overcome with a more sophisticated definition of α , for example, by using a convolution operator. Also note we are using the convention $\tilde{p}_i = 0$ for $i < 1$ or $i > N$.

Now we solve the vector differential equation

$$\dot{x} = A(x)x + \alpha(t)b, \quad (C3)$$

where

$$x = \begin{bmatrix} \tilde{I}_L \\ \tilde{V}_{C_m} \\ \tilde{V}_{C_t} \end{bmatrix} \quad A(x) = \begin{bmatrix} -\frac{R}{L} & 0 & \frac{1}{L} \\ 0 & \frac{-1}{R_L C_m} & \frac{1}{R_L C_m} \\ -\frac{1}{C_t} & \frac{1}{R_L C_t} & \frac{1}{R_L C_t} \end{bmatrix} - i\omega_r \mathbb{I} \quad b = \begin{bmatrix} 0 \\ \frac{1}{R_L C_m} \\ \frac{1}{R_L C_t} \end{bmatrix} \quad (C4)$$

with Mathematica 10's function `NDSolve`. By default, this function dynamically chooses the step size and switches between solvers, of both the implicit and explicit time stepping variety, to ensure that the solution is accurate and stable. An interpolating function for $\tilde{I}_L(t)$ is returned. Recalling Equation B4, we sample the real and imaginary parts of $\tilde{I}_L(t)$ at a rate δt to obtain \vec{q} :

$$\begin{aligned} q_{m,1} &= \kappa \operatorname{Re} \tilde{I}_L(\delta t(m - 1/2)) \\ q_{m,2} &= \kappa \operatorname{Im} \tilde{I}_L(\delta t(m - 1/2)) \end{aligned} \quad (C5)$$

2. Jacobian of the Non-linear Resonator Distortion

To populate elements of the Jacobian tensor $J_{\vec{p}}(g)$, we are interested in approximating partial derivatives of the form

$$\frac{\partial g_{m,l}}{\partial p_{n,k}} \quad (C6)$$

where g is the distortion corresponding to the non-linear resonator circuit. The most straight forward way of approximating such would be to use a central difference formula

$$\frac{\partial g_{m,l}}{\partial p_{n,k}} \approx \left[\frac{g(\vec{p} + \epsilon \vec{e}_{n,k}) - g(\vec{p} - \epsilon \vec{e}_{n,k})}{2\epsilon} \right]_{m,l}, \quad (C7)$$

where $\vec{e}_{n,k}$ is the unit vector in the $\{n, k\}$ direction, and $\epsilon > 0$ is a small number that is greater than the precision of the DE solver. Such an approximation would require $2NK$ calls to the DE solver. It is also numerically unstable as it involves the difference of two numerical DE solutions whose forcing terms are only slightly different; ϵ would have to be very carefully tuned and may have no reliable value at all, especially when searching for high fidelity pulses.

If we consider the approximation $g(\vec{p} \pm \epsilon \vec{e}_{n,k}) \approx g(\vec{p}) \pm g(\epsilon \vec{e}_{n,k})$ the central difference reduces to

$$\frac{\partial g_{m,l}}{\partial p_{n,k}} \approx [g(\epsilon \vec{e}_{n,k})/\epsilon]_{m,l}. \quad (C8)$$

which is the approximation quoted in the main body. Importantly, this approximation does not depend on the current pulse \vec{p} and can therefore be precomputed eliminating the $2NK$ calls to g (i.e. DE solver calls) per ascension step.

An exact method to compute these partial derivatives is derived below, which will take $N * K + 1$ calls to the DE solver to compute the entire Jacobian matrix. Begin with the resonator differential equation (Equation C3)

$$\dot{x} = A(x)x + \alpha(t)b. \quad (C9)$$

As discussed, we have

$$\begin{aligned} [g(\vec{p})]_{m,1} &= \kappa \operatorname{Re} \tilde{I}_L(t_m) \equiv h_1(x(t_m)) \\ [g(\vec{p})]_{m,2} &= \kappa \operatorname{Im} \tilde{I}_L(t_m) \equiv h_2(x(t_m)) \end{aligned} \quad (C10)$$

where $t_m = (m - 1/2)\delta t$. Thus it is clear that the difficult part of computing $\frac{\partial g_{m,l}}{\partial p_{n,k}}$ is computing $\frac{\partial \tilde{I}_L}{\partial p_{n,k}}$, or more generally $\frac{\partial x}{\partial p_{n,k}}$.

We derive a set of $K * N = 2N$ secondary partial differential vector equations whose time sampled solutions produce the necessary partial derivatives. To do this we just take the partial derivative $\frac{\partial}{\partial p_{n,k}}$ of Equation C3, which gives as the l^{th} component of the $(n, k)^{\text{th}}$ equation

$$\frac{\partial}{\partial p_{n,k}} \frac{\partial x_l}{\partial t} = \frac{\partial A_{l,l'}}{\partial x_{l'}} \frac{\partial x_{l'}}{\partial p_{n,k}} x_{l'} + [A(x)]_{l,l'} \frac{\partial x_{l'}}{\partial p_{n,k}} + T_{n,k} b_l \quad (C11)$$

where Einstein summation notation is used and (in the case $\tau_r = 0$)

$$T_{n,k}(t) = \begin{cases} 0 & 0 \leq t \leq dt \\ \vdots & \\ \delta_{1,k} + i\delta_{2,k} & (n-1)dt \leq t \leq ndt \\ \vdots & \\ 0 & 0 \leq t \leq Ndt \end{cases} \quad (C12)$$

Denote

$$\begin{aligned} y_{n,k}(t) &= \frac{\partial x}{\partial p_{n,k}}(t) \\ [A'(x)]_{l,l'} &= \frac{\partial A_{l,l'}}{\partial x_{l'}} x_{l'} \end{aligned} \quad (C13)$$

and commuting the partial derivatives, the components Equation C11 can be rewritten as the non-linear vector PDE

$$\dot{y}_{n,k} = [A'(x) + A(x)]y_{n,k} + T_{n,k}(t)b \quad (C14)$$

where $x(t)$ is the solution to Equation C3. Therefore once $x(t)$ has been computed, we can plug it into each of the DEs for $y_{n,k}$, solve them with the initial condition $y_{n,k}((n-1)dt) = 0$ (by causality $y_{n,k} = 0$ for $t < (n-1)dt$) and we arrive at the exact formula

$$\begin{aligned} \frac{\partial g_{m,l}}{\partial p_{n,k}} &= \frac{\partial h_l(x(t))}{\partial p_{n,k}} \Big|_{t=t_m} \\ &= \frac{\partial h_l}{\partial x_{l'}} \frac{\partial x_{l'}}{\partial p_{n,k}} \Big|_{t=t_m} \\ &= \frac{\partial h_l}{\partial x_{l'}} [y_{n,k}(t_m)]_{l'} \end{aligned} \quad (C15)$$

where h_l was defined implicitly in Equation C10 and each $\frac{\partial h_l}{\partial x_{l'}}$ is easy to compute.

If we take the Taylor series of $A(x)$ about $x = 0$, we have

$$A(x) = A_0 + A_1(x) + A_2(x) + \dots \quad (C16)$$

where each A_p is a matrix polynomial in the coordinates of x with all terms having order exactly p . The 0th order approximation of Equation C14 gives

$$\dot{y}_{n,k} = A_0 y_{n,k} + T_{n,k}(t)b. \quad (\text{C17})$$

In this form we see that $y_{n,k}$ is just the same as x where the DE for x , Equation C14, has been linearized and the forcing is the top hat $T_{n,k}$: $y_{n,k} = x|_{A=A_0, \alpha=T_{n,k}}$. The linearization condition $A = A_0$ is approximately the same as the guarantee $\|A(x) - A_0\| \ll 1$, which can be met by setting $\alpha = \epsilon T_{n,k}$ with ϵ chosen so that $\left\| A\left(\frac{\epsilon \|b\|}{\|A_0\|}\right) - A_0 \right\| \ll 1$. Therefore the zeroth order approximation to the Jacobian is

$$\frac{\partial g_{m,l}}{p_{n,k}} \approx \frac{g(\epsilon e_{n,k})}{\epsilon}. \quad (\text{C18})$$

which is a somewhat more satisfying derivation of Equation C8.

Appendix D: Ringdown Compensation

A resonator or cavity with a large quality factor Q will store energy for times that are long compared to the time steps that are used in pulse design. If this effect is not included in optimization by integrating the distortion differential equation for a sufficient period, then the integrated action of the pulse on the quantum system will not be accurate. This can be dealt with by defining the image of the distortion operator to represent a longer time interval than the domain, but this is inconvenient in experimental practice, where we would like to turn off a pulse quickly. Thus, a better alternative is to actively compensate for the ringdown introduced by large Q , and to demand that the distorted pulse goes to zero at a given time step.

For a resonator with only linear elements, this problem has been solved [21] by appealing to the transfer function $h : \mathbb{R}^M \rightarrow \mathbb{R}^K$,

$$g[\vec{p}] = f_1[f_2(\vec{p}) \star h] \quad (\text{D1})$$

where \star is the convolution operator. For the case $M = K = 1$, the transfer function takes on the simple form

$$h(t) = Ae^{-t/\tau_c} \quad (\text{D2})$$

for some amplitude A and where $\tau_c = Q/\omega_0$ is a time constant. In this case, it is easy to append an additional pulse segment of amplitude

$$p_{K+1} = -A \frac{g[\vec{p}]_m}{e^{\delta t/\tau_c} - 1}, \quad (\text{D3})$$

where m is a time step index such that $t_m = t_K$.

In the nonlinear case, Q , ω_0 and A are not constant, but depend on \vec{p} , and so more attention is required. One solution is to modify the performance functional to include the demand that the ringdown go to zero by defining

$$\Phi'_g(\vec{p}) := \Phi_g(\vec{p}) - \Omega_g(\vec{p}) = (\Phi - \Omega) \circ g. \quad (\text{D4})$$

For ringdown compensation,

$$\Omega_{\text{RD}} := \sum_{m=m_0}^M |p_m|^2, \quad (\text{D5})$$

where m_0 is the time step index at which we start demanding that the solution goes to zero. The derivatives of this function are easily found, such that $\vec{\nabla} \Phi'$ is easy to compute given $\vec{\nabla} \Phi$ and $J(g)$. Since a solution that both has high fidelity with a unitary target and admits ringdown compensation can be hard to find, we use the ringdown-compensation method found in the next section to generate *initial guesses* which result in a small penalty $\Phi'_g(\vec{p})$.

Another solution is to include ringdown suppression in the distortion operator g itself. That is, given an input pulse \vec{p} , the forcing term α now includes not only steps taken directly from \vec{p} , but also additional steps which are chosen (according to the results from the next section) to eliminate the energy from the cavity in a short period of time. This was the method employed for the results shown in the main body of this Letter.

1. Eliminating energy from a non-linear resonator

Here, we derive a scheme to calculate the values of compensation steps to append to a pulse which act to remove the energy from a resonator on a timescale shorter than the ringdown time.

Write the equation of the circuit as

$$\dot{x} = Ax + \alpha b \quad (\text{D6})$$

where x is a vector of state variables for the circuit, A is a matrix describing the circuit without forcing, b is the forcing direction of the circuit, and α is a controllable scalar which sets the magnitude of the forcing. We assume that we have already entered the frame rotating at the resonance frequency so that all quantities are complex, where real quantities correspond to in-phase components, and imaginary quantities correspond to quadrature components. Note that for a non-linear circuit, A will depend on the state of the system, that is, $A = A(x)$. Moreover, α can be time dependent, $\alpha = \alpha(t)$.

Our goal is as follows: start with an undistorted pulse \vec{p}_0 and append n_{rd} steps of length dt_{rd} to form the undistorted pulse $\vec{p} = [\vec{p}_0, \vec{p}_{\text{rd}}]$ which cause the distorted pulse $g(\vec{p})$ to have near zero amplitude at the end of the last time step. To simplify our task, we make the approximation that A remains constant during each of the compensation steps, taking on a value corresponding to the state x at the end of the previous time step.

The general solution to [D6](#) is given by

$$x(t) = e^{tA}x_0 + \int_0^t \alpha(s)e^{(t-s)A}b \, ds. \quad (\text{D7})$$

Substituting our continuous forcing solution from equation [C2](#) and translating the time coordinate so that $t = 0$ corresponds to the transition from the $(n-1)^{\text{th}}$ to the n^{th} gives the solution

$$\begin{aligned} x(t) &= e^{tA}x_0 + e^{tA} \left[\int_0^t e^{-sA} \left(\tilde{p}_{n-1} + (\tilde{p}_n - \tilde{p}_{n-1})(1 - e^{-s/\tau_r}) \right) ds \right] b \\ &= e^{tA}x_0 + \left[\tilde{p}_n A^{-1}(e^{tA} - \mathbb{I}) - (\tilde{p}_n - \tilde{p}_{n-1})(A + \mathbb{I}/\tau_r)^{-1} \left(e^{tA} - e^{-t/\tau_r} \mathbb{I} \right) \right] b \end{aligned} \quad (\text{D8})$$

in the region $t \in [0, dt_{\text{rd}}]$. We wish to drive the state of the system, x , to 0. Therefore, let's try to demand that at time $t = dt_{\text{rd}}$, x becomes some fraction of its value at the end of the $(n-1)^{\text{th}}$ step, so that $x(dt_{\text{rd}}) = rx_0$ for some $r \in [0, 1]$. We refrain from setting $r = 0$ when x is large because if x changes too much in the time span dt_{rd} our approximation of constant A will break down. Since all we can do is change the value of \tilde{p}_n , the equality $x(dt_{\text{rd}}) = rx_0$ won't in general be achievable. We therefore instead minimize the quantity

$$\beta(\tilde{p}_n) = \|P(x(dt_{\text{rd}}) - rx_0)\|_2 \quad (\text{D9})$$

where P is a positive semi-definite matrix which relates the importance of minimizing certain state variables over others. This quantity can be rewritten as

$$\begin{aligned} \beta(\tilde{p}_n) &= \|w - \tilde{p}_n v\|_2 \\ w &= P \left[(e^{tA} - r\mathbb{I})x_0 + \tilde{p}_{n-1} (A + \mathbb{I}/\tau_r)^{-1} \left(e^{tA} - e^{-t/\tau_r} \mathbb{I} \right) b \right] \\ v &= P \left[(A + \mathbb{I}/\tau_r)^{-1} \left(e^{tA} - e^{-t/\tau_r} \mathbb{I} \right) - A^{-1}(e^{tA} - \mathbb{I}) \right] b \end{aligned} \quad (\text{D10})$$

In this form it is clear that $\beta(\tilde{p}_n)$ is minimized when \tilde{p}_n is chosen to be the complex projection amplitude of the vector w onto v :

$$\tilde{p}_n = \frac{\langle v, w \rangle}{\langle v, v \rangle}. \quad (\text{D11})$$

For reference, note that in the limit $\tau_r \rightarrow 0$, the vectors v and w simplify to

$$\begin{aligned} w &= P(e^{tA} - r\mathbb{I})x_0 \\ v &= -PA^{-1}(e^{tA} - \mathbb{I})b. \end{aligned} \quad (\text{D12})$$

Algorithm 1 Modified GRAPE algorithm.

Input: Target unitary U , target fidelity Φ_{target} , distortion operator g , initial pulse \vec{p}_{init} , ringdown compensation steps n_{steps} , ringdown compensation step width τ_r , ringdown compensation ratio $r \in [0, 1]$, [optional] list of samples $\{\vec{x}_i\}_{i=1}^n$.

Output: Pulse \vec{p} such that $\Phi_g[\vec{p}] \geq \Phi_{\text{target}}$, or $\bar{\Phi}_g[\vec{p}] \geq \Phi_{\text{target}}$ if a list of samples is given.

```

function UTIL( $\vec{q}, \{\vec{x}_i\}_{i=1}^n$ )
  return  $\sum_{i=1}^n \Phi[\vec{q}|\vec{x}_i]/n$ 
end function

function RINGDOWNCOMPENSATE( $\vec{q}, x_0, n_{\text{steps}}, \tau_r, r$ )
  for  $i_{\text{step}} \in \{1, \dots, n_{\text{steps}}\}$  do
     $q_0 \leftarrow$  last step in  $\vec{q}$ 
     $w \leftarrow P \left[ (e^{tA} - r\mathbb{I})x_0 + q_0 (A + \mathbb{I}/\tau_r)^{-1} (e^{tA} - e^{-t/\tau_r}\mathbb{I}) b \right]$ 
     $v \leftarrow P \left[ (A + \mathbb{I}/\tau_r)^{-1} (e^{tA} - e^{-t/\tau_r}\mathbb{I}) - A^{-1}(e^{tA} - \mathbb{I}) \right] b$ 
    append  $\langle v, w \rangle / \langle v, v \rangle$  to  $\vec{q}$ 
  end for
  return  $\vec{q}$ 
end function

function FINDPULSE( $U, \Phi_{\text{target}}, g, \vec{p}, n_{\text{steps}}, \tau_r, r, [\{\vec{x}_i\}_{i=1}^n]$ )
  if distribution samples  $\{\vec{x}_i\}_{i=1}^n$  are not given then
     $\{\vec{x}_i\} \leftarrow \{\vec{0}\}$  ▷ Use a single sample if no samples are given.
  end if
   $\beta \leftarrow 0$ 
   $\vec{d}' \leftarrow \vec{0}$ 
   $\vec{g} \leftarrow \sum_{i=1}^n g[\cdot|\vec{x}_i]/n$ 
   $J_g \leftarrow J(\vec{g})$  ▷ Precalculate the Jacobian of the distortion operator  $g$ .
   $u \leftarrow 0$ 
  while  $u \leq \Phi_{\text{target}}$  do
     $\vec{q}, x_0 \leftarrow \vec{g}[\vec{p}]$  ▷ Distort the pulse, keeping the final state  $x_0$  of the distortion.
     $\vec{q} \leftarrow$  RINGDOWNCOMPENSATE( $\vec{q}, x_0, n_{\text{steps}}, \tau_r, r$ ) ▷ Compensate the pulse for energy removal.
     $u \leftarrow$  UTIL( $\vec{q}$ )
     $\vec{d} \leftarrow \sum_{i=1}^n \vec{\nabla}_{\vec{q}} \Phi[\vec{q}|\vec{x}_i] \cdot J_g$  ▷ Use [11] to calculate  $\vec{\nabla}_{\vec{q}} \Phi$ .
     $\Delta \vec{d} \leftarrow \vec{d} - \vec{d}'$  ▷ Find the conjugate gradient direction.
     $\beta \leftarrow \max\{0, \vec{d} \cdot \Delta \vec{d} / \vec{d}' \cdot \vec{d}'\}$ 
     $\vec{s} \leftarrow \vec{d} + \beta \vec{d}'$ 
     $\alpha = \arg \max_{\alpha} \text{UTIL}(\vec{g}[\vec{p} + \alpha \vec{s}], \{\vec{x}_i\})$  ▷ Perform a line search in the “good” direction.
     $\vec{p} \leftarrow \vec{p} + \alpha \vec{s}$  ▷ Update the pulse by the step  $\alpha \vec{s}$ .
     $\vec{d}' \leftarrow \vec{d}$  ▷ Set the previous gradient to the current and prepare for the next iteration.
  end while
  return  $\vec{p}$ 
end function

```

Appendix E: Pseudocode for Modified GRAPE

In this Section, we list our modifications to GRAPE for use with the non-linear resonator as [Algorithm 1](#).

Appendix F: Static Parameter Distributions

We discuss a well known and somewhat trivial modification to the GRAPE algorithm that deals with uncertainties in physical parameters which are static with respect to the length of a single shot of the experiment, whether the single shot be spacial or temporal. The classic example is a small inhomogeneity in the static field of an NMR magnet; up to diffusion, the molecules stay fixed in space and therefore each spin will have a slightly different resonance frequency corresponding to the value of the magnetic field at its position.

If the distortion or Hamiltonian is not known precisely, but instead follows a distribution, we can consider the *conditional* performance function, $\Phi_g[\vec{p}|\vec{x}, \{H\}_{i=0}^L] = \Phi(g(\vec{p}, \vec{x})|\{H\}_{i=0}^L)$, where we have allowed g to be a function of an additional vector \vec{x} and made the dependence of Φ on the system and control Hamiltonians explicit [11]. We are then interested in maximizing the marginalized objective function, $\bar{\Phi}_g[\vec{p}] := \mathbb{E}_{\vec{x}, \{H\}_{i=0}^L} [\Phi_g[\vec{p}|\vec{x}, \{H\}_{i=0}^L]]$. Since the

expectation operator is linear, this implies that we can find the gradients of the marginalized objective function by averaging over the gradients of the conditional objective function, $\nabla_{\vec{p}}(\bar{\Phi}_g) = \mathbb{E}[\nabla_{g(\vec{p})}(\Phi(g(\vec{p}, \vec{x})|\{H\}_{i=0}^L) \cdot J_{\vec{p}}(g, \vec{x})]$. Numerically, it is convenient to approximate this expectation value by maintaining a list of hypothesis about \vec{x} and $\{H\}_{i=0}^L$.